UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/670,916	09/29/2000	Mariusz H. Jakubowski	MS1-527US	1893

22801 7590 04/07/2005

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER	
JACKSON, JENISE E	
ART UNIT	PAPER NUMBER
2131	

DATE MAILED: 04/07/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/670,916	JAKUBOWSKI ET AL.	
	Examiner	Art Unit	
	Jenise E Jackson	2131	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-44* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-44* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

2. Claims 1-44 are rejected under 35 U.S.C. 102(e) as being anticipated by Rump et al.(6735311).

3. As per claim 1, Rump et al. discloses identifying a plurality of modules in a software program, wherein each module includes a plurality of blocks (see col. 2, lines 1-26), because Rump discloses multimedia data(i.e. software) which area(i.e. module) assigned specific definition data blocks(see col. 7, lines 22-25); the multimedia(software) is subdivided into separate data blocks(see col. 7, lines 22-36); and wherein the plurality of modules includes checker modules(i.e. checksum), each data block contains a checksum, because Rump discloses that the checksum of the definition data block is itself ciphered, an no unauthorized person will be able to recreate the data block(see col. 8, lines 48-51, see fig. 1, sheet 1); for each of the plurality of modules, generating an original checkpoint value, and incorporating the original checkpoint value into a checker module(is the data block that contains the checksum value)(col. 6, lines 38-60); and for each checker modules, generating a new checkpoint value after the original checkpoint value has been incorporated into the checker module(see col. 8, lines 25-37), and determining a new block to add to the checker module to offset the incorporated original

checkpoint value such that subsequent generation of a checkpoint value for the checker module equals the original checkpoint value for the checker module(see col. 6, lines 61-67, col. 8, lines 25-61), Rump discloses this because Rump discloses a free index field that is added to a checker module. The free index field of Rump provides an assignment to the data block to the ciphered(checksum) multimedia data(see col. 8, lines 38-61). When deciphering the multimedia data a check is made to see whether the checksum calculated from the data agrees with the subentry user data. This user data contains the free index filed of Rump(see col. 8, lines 38-43, 54-61).

4. As per claim 2, Rump discloses incorporating the original checkpoint value into multiple checker modules(see col. 7, lines 18-35).

5. As per claim 3, Rump discloses computing, based on the plurality of blocks of a module, a message authentication code(MAC) value to be used as the checkpoint value for the module(see col. 8, lines 46-61).

6. As per claim 4, Rump discloses inputting each of the plurality of blocks of the module into an exclusive-or operation on each block and an encrypted version of the previous output of the exclusive-or operator(see col. 6, lines 26-37); and using, as the message authentication code value, the output value from the exclusive-or operator obtained from inputting the last of the plurality of blocks into the exclusive-or operator(see col. 8, lines 25-37).

7. As per claim 5, Rump discloses wherein the determining a new block, encrypting the new checkpoint value(see col. 2, lines 27-45); and determining, as the content of the new block, a value equal to the exclusive-or of the encrypted new checkpoint value and the original checksum value(see col. 2, lines 46-67).

8. As per claim 6, Rump discloses wherein the new block does not alter the functionality of the module(see col. 3, liens 5-25).

9. As per claim 7, Rump discloses wherein the new block includes a data block(see col. 5, lines 16-25).

10. As per claim 8, Rump discloses wherein the plurality of instructions, when executed, further cases the one or more processors to perform acts including adding, prior to generating the new checkpoint value(see col. 6, lines 38-60), additional instructions to the module as part of one or more additional blocks, the additional instructions causing the addition of the new block to not alter the functionality of the module(see col. 10, lines 57-67).

11. As per claim 9, Rump discloses wherein the software program includes a plurality of checkpoints corresponding to the incorporated checkpoint values(see col. 6, lines 38-60), wherein each checkpoint identifies when the integrity of the corresponding module is to be verified(see col. 10, lines 57-67).

12. As per claims 10, 14, Rump discloses identifying a plurality of segments in an object; and applying cyclic integrity verification to the object based on the plurality of segments(see col. 6, lines 38-60).

13. As per claim 11, Rump discloses wherein the cyclic integrity verification is applied to the plurality of segments by: for each of the plurality of segments, generating an original checkpoint value, and incorporating the original checkpoint value into a checker segment(see col. 6, lines 38-60, col. 10, lines 57-67); and for each of the checker segments, generating a new checkpoint value after the original checkpoint value has been incorporated into the checker segment(see col. 6, lines 61-67, col. 8, lines 25-37), determining an additional block to be added to the checker

segment to offset the incorporated original checkpoint value such that subsequent generation of a checkpoint value for the checker segment equals the original checkpoint value for the checker segment(col. 8, lines 25-37).

14. As per claim 12, Rump discloses wherein the cyclic integrity verification is applied to verify the shape of the plurality of segments(see col. 6, lines 38-60).

15. As per claim 13, Rump discloses wherein the cyclic integrity verification is applied to verify the behavior of the plurality of segments(see col. 10, lines 57-67).

16. As per claims 15, 24, Rump discloses identifying a plurality of segments in an object; generating a checkpoint value for each of the plurality of segments(see col. 6, lines 38-60); storing the checkpoint value for each of the plurality of segments in another of the plurality of segments(see col. 2, lines 3-26); and modifying each of the plurality of segments so that the addition of the checkpoint value to the segment is offset and the checkpoint value for the segment remains the same(see col. 2, lines 27-66).

17. As per claim 16, Rump discloses wherein the storing the checkpoint value into multiple other segments of the plurality of segments(see col. 3, lines 5-25).

18. As per claim 17, Rump discloses computing, based on a plurality of blocks of a segment, a message authentication code(MAC) value to be used as the checkpoint value for the segment(see col. 8, lines 46-61); and determining a new block to add to the segment to offset the stored checkpoint value such that subsequent generation of a checkpoint value for the segment equals the previously generated message authentication code value(col. 8, lines 25-37).

19. As per claim 18, Rump discloses inputting each of the plurality of blocks of the segment into an exclusive-or operator that generates an output value by performing an exclusive-or

operation on each block(see col. 6, lines 26-37) and encrypted version of the previous output of the exclusive-or operator; and using, as the message authentication code value(see col. 4, lines 62-67), the output value from the exclusive-or operator obtained from inputting the last of the plurality of blocks into the exclusive-or operator(see col. 9, lines 32-59).

19. As per claim 19, Rump discloses generating a new checkpoint value based on the plurality of blocks and including the stored checkpoint value(see col. 6, lines 38-60); encrypting the new checkpoint value; and determining, as the content of the new block, a value equal to the exclusive-or of the encrypted new checkpoint value and the original checksum value(see col. 2, lines 27-66).

20. As per claim 20, Rump discloses wherein the modifying does not alter the functionality of the segment(see col. 8, lines 4-15).

21. As per claim 21, Rump discloses wherein the modifying includes adding a new data block(see col. 5, lines 16-25).

22. As per claim 22, Rump discloses wherein the object includes a software program(see col. 3, lines 25-36, col. 4, lines 19-25).

23. As per claim 23, Rump discloses storing a checkpoint corresponding to each checkpoint value, each checkpoint identifying when the integrity of the corresponding segment is to be verified (see col. 10, lines 57-67).

24. As per claims 25, 37, Rump discloses generating a verification value for a first segment of an object(see col. 6, lines 38-60); generating an original verification value for a second segment of the object; adding, to the second segment, the verification value for the first segment and adding an offset value to the second segment so that a newly calculated verification value for

the second segment equals the original verification value(see col. 8, lines 46-61, col. 10, lines 19-28).

25. As per claim 26, Rump discloses wherein the generating the verification value for the first segment includes generating the verification value based at least in part on behavior of the first segment during execution of the first segment(see col. 2, lines 3-26).

26. As per claim 27, Rump discloses wherein the behavior of the first segment during execution includes modification of a register by one or more instructions in the first segment during execution(see col. 8, lines 4-37).

27. As per claim 28, Rump discloses adding, to the first segment, the original verification value for the second segment; and adding another offset value to the first segment so that a newly calculated verification value for the first segment equals the verification value for the first segment(see col. 6, lines 61-67, col. 7, lines 1-13).

28. As per claim 29, Rump discloses wherein generating the original verification value includes computing, based on a plurality of blocks of the second segment, a message authentication code(MAC) value(see col. 8, lines 46-61).

29, As per claim 30, limitations have already been addressed(see claim 18).

30. As per claim 31, Rump discloses generating a new verification value for the second segment(see col. 6, lines 26-37); encrypting the new verification value; and determining, as the offset value, a value equal to the exclusive-or of the encrypted new verification value and the original verification value(see col. 8, lines 25-37, col. 9, lines 32-58).

31. As per claim 32, Rump discloses wherein the offset value does not alter the functionality of the module(see col. 6, lines 38-60).

33. As per claim 33, Rump discloses wherein the offset value includes a data block(see col. 5, lines 16-25).

34. As per claim 34, Rump discloses wherein the object includes a software program(see col. 3, lines 25-36, col. 4, lines 19-25).

35. As per claim 35, Rump discloses storing a checkpoint, corresponding to the verification value, that identifies when the integrity of the first segment is to be verified (see col. 6, lines 38-60, col. 8, lines 46-61).

36. As per claim 36, Rump discloses storing the checkpoint in the second segment(see col. 8, lines 46-61).

37. As per claim 38, Rump discloses a plurality of segments, each including one or more checkpoint values to be used to verify the integrity of one or more other segments(see col. 5, lines 46-62); and wherein the plurality of segments further include a plurality of checkpoints that identify a circular ordering of verifying the integrity of the segments(see col. 6, lines 61-67, col. 8, lines 25-37).

38. As per claim 39, Rump discloses wherein each of the checkpoint values is message authentication code value based on the one or more other segments(see col. 6, lines 38-60, col. 8, lines 46-61).

39. As per claim 40, Rump discloses wherein each of the plurality of segments includes a checkpoint value to be used to verify the integrity of each of the other of the plurality of segments(see col. 6, lines 38-60, col. 8, lines 46-61).

40. As per claim 41, Rump discloses a memory to store an original program; and a production server equipped with a cyclic integrity verification protection tool that is used to

augment the original program for protection purposes(see col. 5, lines 7-25), the production server being configured to parse the original program into a plurality of segments and apply cyclic integrity verification to the plurality of segments(see col. 7, lines 18-36).

41. As per claim 42, limitations already addressed(see claim 11).

42. As per claim 43, Rump discloses wherein the cyclic integrity is applied to the plurality of segments by including a plurality of checkpoints corresponding to the incorporated checkpoint values(see col. 6, lines 61-67, col. 8, lines 25-37), wherein each checkpoint identifies when the integrity of the corresponding segment is to be verified(col. 6, lines 38-60).

43. As per claim 44, Rump discloses a production server to apply cyclic integrity verification to a program to produce a protected; and a client to store and execute the protected program, the client being configured to evaluate the protected program to determine whether the protected program has been tampered with(see col. 5, lines 46-67, col. 6, lines 6-15).

Response to Amendment

44. The Applicant states that Rump does not disclose an original checkpoint value is generated for each of a plurality of modules in a software program, and this original checkpoint value is incorporated into a checker module. Also, the Applicant states that Rump does not disclose a new checkpoint value is generated for each of the checker modules after the original checkpoint value has been incorporated into the checker module. The Examiner disagrees with the Applicant. Rump discloses in the block of the free index, a checksum is included. When a reproduction, a deciphering is enacted, a check is made to see whether the checksum value calculated from agrees with the entry of the user data field(see col. 8, lines 46-61).

45. The Applicant states that Rump does not disclose cyclic integrity verification. The Examiner disagrees with the Applicant. Rump does disclose a cyclic integrity verification, because Rump discloses checking the integrity of multimedia data, by using a checksum(see col. 6, lines 39-60).

46. In regards to Applicant's remarks located on pages 22-23, Applicant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references.

47. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jenise E Jackson whose telephone number is (571) 272-3791. The examiner can normally be reached on M-Th (6:00 a.m. - 3:30 p.m.) alternate Friday's.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ayaz Sheikh can be reached on (571) 272-3795. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

April 1, 2005

AYAZ SHEIKH
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100